CLAIMS

What is Claimed is:

1. A method comprising:

stalling a call to a critical operating system (OS) function; and

determining whether branch trace records of said call include a return instruction.

2. The method of Claim 1 wherein said determining whether branch trace records of said call include a return instruction comprises:

locating a most recent branch trace record of said branch trace records;

searching said branch trace records from said most recent branch trace record to locate a user to kernel branch trace record of said branch trace records; and

searching previous branch trace record previous to said user to kernel branch trace record for said return instruction.

3. The method of Claim 2 wherein upon a determination that said previous branch trace records do not include said return instruction, said method further comprising allowing said call to proceed.

4. The method of Claim 2 wherein upon a determination that at least one of said previous branch trace records does include said return instruction, said method further comprising taking protective action to protect a computer system.

5. The method of Claim 1 further comprising taking protective action to protect a computer system upon a determination that said branch trace records include said return instruction.

6. The method of Claim 5 wherein said taking protective action comprises terminating said call.

7. The method of Claim 5 wherein said taking protective action comprises terminating a call module originating said call.

8. The method of Claim 5 wherein said taking protective action comprises terminating a parent application comprising a call module originating said call.

9. The method of Claim 5 further comprising providing a notification that said protective action has been taken.

10. The method of Claim 1 further comprising allowing said call to proceed upon a determination that said branch trace records do not include said return instruction.

11. The method of Claim 1 wherein upon a determination that said branch trace records include said return instruction, said method further comprising determining whether said call is a known false positive.

12. The method of Claim 11 wherein upon a determination that said call is not said known false positive, said method further comprising taking protective action to protect a computer system.

13. The method of Claim 12 further comprising providing a notification that said protective action has been taken.

14. The method of Claim 11 wherein upon a determination that said call is said known false positive, said method further comprising allowing said call to proceed.

15. The method of Claim 1 further comprising hooking said critical OS function.

16. The method of Claim 1 further comprising recording said branch trace records.

17. The method of Claim 16 further comprising suspending recording of said branch trace records prior to said determining whether branch trace records of said call include a return instruction.

18. The method of Claim 17 further comprising unsuspending recording of said branch trace records after said determining whether branch trace records of said call include a return instruction.

19. The method of Claim 1 wherein said critical OS function is necessary for a first application to cause execution of a second application.

20. The method of Claim 19 wherein said second application allows remote access of a computer system.

21. A method comprising:

recording branch trace records;

stalling a call to a critical operating system (OS) function;

suspending recording of said branch trace records;

locating a most recent branch trace record of said branch trace records;

searching said branch trace records from said most recent branch trace record to locate a user to kernel branch trace record of said branch trace records; and

determining whether previous branch trace records previous to said user to kernel branch trace record include only call, jump, or interrupt instructions.

22. The method of Claim 21 wherein said determining whether previous branch trace records previous to said user

to kernel branch trace record include only call, jump, or interrupt instructions is performed until a determination is made that a last branch trace record has been reached.

23. The method of Claim 22 wherein upon a determination that said last branch trace record has been reached, said method further comprising allowing said call to proceed.

24. The method of Claim 21 wherein said determining whether previous branch trace records previous to said user to kernel branch trace record include only call, jump, or interrupt instructions is performed until a determination is made that at least one of said previous branch trace records includes a return instruction.

25. The method of Claim 24 wherein upon a determination that said at least one of said previous branch trace records includes said return instruction, said method further comprising taking protective action to protect a computer system.

26. A computer program product comprising:

a Return-to-LIBC attack detection application for recording branch trace records;

said Return-to-LIBC attack detection application further for stalling a call to a critical operating system (OS) function;

said Return-to-LIBC attack detection application further for suspending recording of said branch trace records;

said Return-to-LIBC attack detection application further for locating a most recent branch trace record of said branch trace records;

said Return-to-LIBC attack detection application further for searching said branch trace records from said most recent branch trace record to locate a user to kernel branch trace record of said branch trace records; and

said Return-to-LIBC attack detection application further for determining whether previous branch trace records previous to said user to kernel branch trace record include only call, jump, or interrupt instructions.